

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representation of
The original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

=> s de4433013/pn
L6 1 DE4433013/PN

=> d ab

L6 ANSWER 1 OF 1 WPINDEX COPYRIGHT 2002 DERWENT INFORMATION LTD
AB WO 9608753 A UPAB: 19960510

The control and activation system is used for sensor s(6) and/or actuators linked via bus and driven by a host computer (2) via an ASI master (1). The host computer is integrated in the ASI master as an autonomous computer, with a common controller for implementing the ASI master software and the host computer software. Priority is given to the ASI master software, to prevent the timing of the ASI master being altered or interrupted by the host computer or the host computer software, while the ASI master can interrupt the host computer at any time, the ASI data delivered to the ASI master by the host computer software via an internal interface.

USE/ADVANTAGE - For automation system with reduced wiring and control complexity.

Dwg.1/5



① BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENTAMT

⑫ Offenlegungsschrift
⑩ DE 44 33 013 A 1

⑤ Int. Cl.⁶:
G 05 B 9/02
G 05 B 23/02
G 05 B 19/05
G 06 F 11/30

⑳ Aktenzeichen: P 44 33 013.8
㉔ Anmeldetag: 15. 9. 94
㉕ Offenlegungstag: 28. 3. 96

DE 4433013 A 1

⑦① Anmelder:
Bihl, Jochen, 68623 Lampertheim, DE; Wiedemann,
Bernhard, 68161 Mannheim, DE

⑦④ Vertreter:
Mierswa, K., Dipl.-Ing., Pat.- u. Rechtsanw., 68199
Mannheim

⑦② Erfinder:
gleich Anmelder

Prüfungsantrag gem. § 44 PatG ist gestellt

⑤④ Verfahren und Vorrichtung zur Steuerung und Aktivierung von miteinander mittels eines Bussystems vernetzten Sensoren und/oder Aktuatoren

⑤⑦ Die Erfindung betrifft ein Verfahren zur Steuerung und Aktivierung von mittels eines Bussystems vernetzten binären Sensoren und/oder Aktuatoren (ASI-Slaves), die von einem Verarbeitungsrechner (Hostrechner), wie speicherprogrammierbare Steuerung, über ein einen Controller aufweisendes Aktuator-Sensor-Interface (ASI-Master) angesteuert werden und umgekehrt, welches die auf das Bussystem von den ASI-Slaves aufgegebenen Signale des Hostrechners in einem vorgegebenen Zeitraster (ASI-Masterprogramm) zur Verfügung stellt, und der ASI-Master das Bussystem zu jedem Zeitpunkt in einen sicheren Zustand zu versetzen imstande ist. Der Verarbeitungsrechner ist als autarker Rechner innerhalb des ASI-Masters integriert, aber getrennt aufgebaut, wobei das ASI-Masterprogramm vom Programm des Hostrechners entkoppelt ist, jedoch ein Datenaustausch zwischen beiden stattfindet. Das ASI-Masterprogramm besitzt Priorität gegenüber dem Hostprogramm dergestalt, daß zu keinem Zeitpunkt der zeitliche Ablauf des ASI-Masters durch den Hostrechner bzw. das Hostprogramm verändert oder unterbrochen werden kann, hingegen der ASI-Master den Hostrechner zu jedem Zeitpunkt zu unterbrechen imstande ist, und das Hostprogramm seine Daten über ein interne Schnittstelle an den ASI-Master übergibt.

DE 4433013 A 1

Technisches Gebiet

Die Erfindung betrifft ein Verfahren zur Steuerung und Aktivierung von miteinander mittels eines Bussystems vernetzten binären Sensoren und/oder Aktuatoren (ASI-Slaves), die von einem Verarbeitungsrechner (Hostrechner) über ein einen Controller aufweisendes Aktuator-Sensor-Interface (ASI-Master) angesteuert werden gemäß dem Oberbegriff des Patentanspruchs 1 sowie eine Vorrichtung zur Durchführung des Verfahrens.

Stand der Technik

Bisher war es bei Feldbussystemen aus Kosten- und Platzgründen nicht möglich, binäre Sensoren oder Aktuatoren direkt busfähig zu machen. Durch hochintegrierte Technologien können heute Sensoren neben dem eigentlichen Schaltzustand noch weitere Funktionen liefern, die Einstell- und Diagnosemöglichkeiten der Sensoren oder Aktuatoren bieten; diese Funktionen mußten bisher durch zusätzliche Leitungen und damit zusätzlichem Verdrahtungsaufwand realisiert werden. Um diese Nachteile zu beheben, wurde der Aktuator-Sensor-Interface-Standard, ASI-Standart genannt, geschaffen, der ein Feldbuskonzept darstellt, mit dem binäre Aktuatoren und Sensoren mit der untersten bzw. ersten Steuerungsebene verknüpft werden, um sie kommunikationsfähig zu machen. Das Aktuator-Sensor-Interface ersetzt dabei den Kabelbaum, Verteilerschränke, Klemmleisten usw. durch ein einfaches Zweileiter-Flachbandkabel, über das Daten mit den Peripherieelementen ausgetauscht werden und das diese zugleich mit Energie versorgt. Mit einem sogenannten separaten ASI-Anschluß in Form eines standardisierten Moduls, der Teil der Busstruktur ist, macht ASI zunächst einmal die meisten konventionellen Peripherieelemente busanschlußfähig. Beim integrierten ASI-Anschluß befindet sich hingegen in einem Gerät ein sogenannter Slave-Baustein, der dadurch selbst busfähig ist (ASI-Verein in: Sonderdruck aus Feldbussysteme für die Investitionsgüterindustrie, Herausgeber VDMA, Frankfurt 1992, Stand 31. 12. 1992 sowie Druckschrift: Fabrikautomation VariNet-A Aktuator-Sensor-Interface, Katalog Sensorsysteme 5, Ausgabe 1994, Herausgeber: Firma Pepperl + Fuchs GmbH, 68301 Mannheim).

Der Master, ASI-Master genannt, übernimmt alle Aufgaben, die für die Entwicklung des Busbetriebs der Slaves notwendig sind einschließlich von Aufgaben der Initialisierung und der Diagnose. Über den ASI-Master ist an den Feldbus ein übergeordneter Verarbeitungsrechner, nämlich Hostrechner, wie speicherprogrammierbare Steuerung oder Bus-Rechner oder PC oder VME-Busrechner, angeschlossen, dem sämtliche Signale aller ASI-Slaves zugeführt werden, wobei der ASI-Master gewährleistet, daß die Signale dem Hostrechner in einem festen Zeitrahmen zur Verfügung gestellt werden und umgekehrt die Steuerungsbefehle des Hostrechners den ASI-Slaves aufgegeben werden. Der ASI-Master stellt außerdem sicher, daß hinzugekommene Slaves erkannt und ausgefallene Slaves an den Hostrechner gemeldet werden; der ASI-Master paßt somit die ASI-Funktionen der Slaves an das externe Verarbeitungssystem des Hostrechners an. Der ASI-Master besitzt für die Bearbeitung dieser Aufgaben normalerweise

seinen Controller, der ein enges Zeitraster einhalten muß. Die Bitzeit bei ASI beträgt 6 µsec, wobei der Controller zyklisch alle 150 µsec ein komplettes ASI-Telegramm auswerten muß. Zu jedem Zeitpunkt kann eine Situation auftreten, bei der der ASI-Master den ASI-Kreis in einen sicheren Zustand versetzen muß. Diese Forderung macht es bisher notwendig, daß der ASI-Master getrennt vom Hostrechner realisiert ist in Form von zwei physikalisch getrennten Geräten, die mit einer bidirektionalen Kommunikationsverbindung verbunden sind. Die Verbindung zwischen Hostrechner und ASI-Master kann zum Beispiel ein Rückwandbus, eine serielle Schnittstelle oder ein beliebiger Feldbus o.a. sein. Dieser Geräteaufwand begrenzt in physikalischer Hinsicht die Einsatzfähigkeit eines derartigen ASI-Masters, da zusätzlich der Hostrechner mit dem ASI-Master verbunden sein muß.

Technische Aufgabe

Der Erfindung liegt die Aufgabe zugrunde, ein Verfahren und eine Vorrichtung der eingangs genannten Gattung anzugeben, mit der der elektrische Aufwand, insbesondere Verdrahtungs- und Steuerungsaufwand, für die Realisierung des Hostrechners und des ASI-Masters vereinfacht werden kann, um einen schnelleren und sicheren Busbetrieb zu gewährleisten.

Offenbarung der Erfindung und deren Vorteile

Die Lösung der Aufgabe besteht beim erfindungsgemäßen Verfahren darin, daß der Verarbeitungsrechner als autarker Hostrechner innerhalb des ASI-Masters integriert (ASI-Steuermaster), aber getrennt aufgebaut ist und das ASI-Masterprogramm vom Programm des Hostrechners (Hostprogramm) entkoppelt ist, jedoch ein Datenaustausch zwischen beiden stattfindet, wobei das ASI-Masterprogramm des ASI-Masters immer Priorität gegenüber dem Hostprogramm des Hostrechners besitzt dergestalt, daß zu keinem Zeitpunkt der zeitliche Ablauf des ASI-Masters durch den Hostrechner bzw. das Hostprogramm verändert oder unterbrochen werden, hingegen der ASI-Master bzw. das ASI-Masterprogramm den Hostrechner bzw. das Hostprogramm zu jedem Zeitpunkt zu unterbrechen imstande ist, und der Hostrechner die Daten des Hostprogramms über eine interne Schnittstelle an den ASI-Master übergibt.

Dieses Verfahren besitzt den Vorteil, daß nunmehr der Hostrechner und der ASI-Master in ein- und demselben Gerät integriert sind, der sogenannte ASI-Steuermaster, weshalb sämtliche Zuleitungen zwischen beiden Geräten weitestgehend entfallen. Erfindungsgemäß erledigt der ASI-Steuermaster Steuerungsaufgaben, für die bei konventioneller Realisierung des Standes der Technik ein Hostrechner zusätzlich erforderlich ist. Dieser ist in der Lage, Aufgaben des Hostrechners und ASI-Masteraufgaben zu übernehmen. Es ist möglich, daß Hostrechner wie ASI-Master sogar auf denselben Controller zurückgreifen, wobei jedoch immer gewährleistet ist, daß das ASI-Master-Programm immer die höhere Priorität gegenüber dem Hostprogramm besitzt. Dadurch können Eingangssignale direkt mit Ausgangssignalen verknüpft werden. Die logische Verbindung verschiedener Eingänge und/oder Ausgänge kann einen Ausgang und/oder Eingang schalten und umgekehrt. Nach dem Start des Programms des Verarbeitungsrechners, ASI-Steuermaster-Programm genannt, kann der

ASI-Steuermaster autark die am ASI-Kreis angeschlossenen Slaves bzw. Anlagen steuern, so daß neben den Aufgaben des ASI-Masters, die mit dem modernen Verkabelungssystem zu vergleichen sind, auch Steuerungsaufgaben der angeschlossenen Anlagen abgearbeitet werden.

Damit die unterschiedlichen Aufgaben des ASI-Masters und des Hostrechners fehlerfrei unabhängig voneinander durchgeführt werden können, werden erfindungsgemäß innerhalb des ASI-Steuermasters die Prozeßdaten zweimal unabhängig voneinander abgelegt und verwaltet, wobei das ASI-Masterprogramm und das Hostprogramm immer nur auf seinen zugewiesenen eigenen Datenbereich zugreifen kann und die Daten der beiden Programme zu vorgebbaren Zeitpunkten ausgetauscht werden. Damit ist gewährleistet, daß das Hostprogramm fehlerfrei arbeitet, auch wenn zu einem beliebigen Zeitpunkt Veränderungen im ASI-Kreis auftreten. Das bedeutet, daß die beiden internen Prozesse, nämlich ASI-Master und Host, eine definierte Schnittstelle besitzen müssen und die Prozesse entkoppelt sind.

Lediglich das ASI-Masterprogramm gibt Daten — und zwar sämtliche Daten — über die ASI-Schnittstelle nach außerhalb des ASI-Steuermasters auf die ASI-Leitung an die ASI-Slaves; der Datenaustausch mit den ASI-Slaves wird vom ASI-Master bedient. Der Hostrechner übergibt die Daten des Hostprogramms an den ASI-Master, der wiederum für das Weiterleiten der Daten und den Datenaustausch mit den ASI-Slaves sorgt. Lediglich der ASI-Master tauscht über eine eigene Schnittstelle nach außerhalb des ASI-Steuermasters Daten mit den ASI-Slaves aus, wobei der Hostrechner die Daten des Hostprogramms mit dem ASI-Master austauscht, der wiederum für den gesamten Datenaustausch auf den ASI-Kreis mit den ASI-Slaves sorgt.

Eine Fehlprogrammierung des Hostprogramms durch einen Anwender des ASI-Steuermasters beeinflusst den Ablauf des ASI-Masters nicht. Gerät das Hostprogramm in eine Endlosschleife durch eine Fehlprogrammierung, dann bleibt davon der ASI-Master unbeeinflusst; der ASI-Kreis wird weiterhin spezifikationskonform betrieben. Wird dem ASI-Master beispielsweise über einen Watchdog gemeldet, daß sich das Steuerprogramm in einer Endlosschleife befindet, dann kann der ASI-Master den ASI-Kreis in einen sicheren Zustand versetzen. Ein sicherer und gefahrloser Zustand bei ASI ist beispielsweise die Offline-Phase. In der Offline-Phase wird zuerst an allen ASI-Slaves ein RESET durchgeführt und danach wird die Kommunikation auf der ASI-Leitung eingestellt. Nach dem RESET und dem Einstellen der Kommunikation besteht keine weitere Gefahr für eine Anlage.

Tritt unabhängig vom Zustand des Hostprogramms beispielsweise ein ASI-Power-Fail, wie Leistungsausfall auf dem Bus, auf, dann wird der ASI-Steuermaster sofort in die Offline-Phase versetzt und zwar unabhängig vom Zustand des Hostprogramms. Fällt ein ASI-Slave aus, dann wird der Slave vom ASI-Master aus der Liste der erkannten Slaves gestrichen und die Daten- und Parameterwerte des Slaves werden beispielsweise mit Defaultwerten beschrieben, was unabhängig vom Zustand des Hostprogramms geschieht.

Des weiteren kann der ASI-Master zu jedem Zeitpunkt einen Slave der Adresse Null auf die Adresse eines fehlenden projektierten Slaves programmieren und somit ein automatisches Programmieren durchführen; das Hostprogramm erlaubt eine Veränderung am realen Prozeß zu jedem Zeitpunkt.

Erkennt der ASI-Master zu einem beliebigen Zeitpunkt einen Fehler, dann werden die entsprechenden ASI-Flags spezifikationskonform beschrieben. Zusätzlich zu den ASI-Flags wird ein globales ASI-Fehlerbit gesetzt, welches nur vom ASI-Master gesetzt werden und nur vom Hostprogramm gelöscht werden darf. Ein vom ASI-Master erkannter Fehler wird für den Hostrechner gespeichert. Durch das Speichern von Fehlern wird eine Konsistenz der Daten beim Datenaustausch zwischen dem Hostrechner und dem ASI-Master erreicht. Der Hostrechner kann auf kurzzeitige ASI-Fehler in einer, im Hostprogramm vorgegebenen Art und Weise, reagieren.

Vor dem Datenaustausch zwischen dem ASI-Masterprogramm und dem Hostprogramm wird überprüft, ob ein Fehler vorliegt. Bei Nichtvorliegen eines Fehlers werden die Daten der beiden Programme ausgetauscht und als neue gültige Daten akzeptiert, hingegen gehen bei Vorliegen eines Fehlers beide Prozesse in die jeweilige Fehlerbehandlung über und verwerfen die ausgetauschten Daten als fehlerhaft. Nach beendetem Datenaustausch wird erneut eine Fehlerprüfung durchgeführt, ob nun aktuell ein Fehler vorliegt oder ob in der Vergangenheit, angezeigt durch das ASI-Fehlerbit, zwischenzeitlich ein Fehler aufgetreten ist. Im Fehlerfall gehen die Prozesse in die Fehlerbehandlung über und verwerfen die ausgetauschten Daten, ohne Fehler werden die übertragenen Daten als gültig akzeptiert und der Datenaustausch ist beendet, wobei bis zum nächsten Datenaustausch beide Prozesse autark arbeiten.

Erfindungsgemäß hat im ASI-Steuermaster der implementierte ASI-Master gegenüber dem implementierten Hostrechner immer und zu jedem Zeitpunkt die höhere Priorität. Der zeitliche Ablauf des ASI-Masters darf zu keinem Zeitpunkt und unter keinen Umständen durch den implementierten Hostrechner verändert, gebremst oder unterbrochen werden. Demgegenüber kann der ASI-Master bzw. das ASI-Masterprogramm den implementierten Hostrechner zu jedem Zeitpunkt unterbrechen; der ASI-Master bzw. das ASI-Masterprogramm ist somit imstande, die Zykluszeit des Hostprogramms zu ändern, die veränderlich sein kann. Reaktionen des internen ASI-Masters auf Fehler werden in einem Zeitraster, vorzugsweise festes Zeitraster, durchgeführt.

Kurzbeschreibung der Zeichnung, in der zeigen:

Fig. 1 die Implementierung eines ASI-Masters und eines Hostrechners in einem ASI-Steuermaster

Fig. 2 die prinzipielle Arbeitsweise des ASI-Steuermasters mittels des implementierten ASI-Masters und des implementierten Hostrechners

Fig. 3 die Aufteilung der Rechenzeit auf die beiden asynchronen Prozesse des ASI-Masters und des Hostrechners bei der Realisierung des ASI-Steuermasters mit einem einzigen Controller

Fig. 4 den zeitlichen Ablauf des Hostprogramms und Fig. 5 ein Hauptdiagramm für den Datenaustausch.

Bevorzugte Ausführungsform der Erfindung

Fig. 1 zeigt die Implementierung eines ASI-Masters 1 und eines Hostrechners 2 innerhalb eines ASI-Steuermasters 3, der über eine Leitung 4 mit dem ASI-Kabel 5 verbunden ist, an dem eine Mehrzahl von ASI-Slaves 6 angebunden sind.

In Fig. 2 ist die prinzipielle Arbeitsweise des ASI-Steuermasters mittels des implementierten ASI-Masters und des implementierten Hostrechners dargestellt.

Der ASI-Master bearbeitet sein ASI-Masterprogramm, der Hostrechner bearbeitet sein Hostprogramm; beide asynchronen Prozesse sind prinzipiell unabhängig voneinander, wobei zwischen beiden Programmen ein Datenaustausch stattfindet, was durch die beide Prozesse verbindende Schleife "Datenaustausch" dargestellt ist. Dabei arbeitet das Hostprogramm fehlerfrei, auch wenn zu einem beliebigen Zeitpunkt Veränderungen im ASI-Kreis auftreten. Der zeitliche Ablauf des ASI-Masters und des Hostrechners verläuft asynchron. Die beiden Prozesse ASI-Master und Hostrechner arbeiten völlig autark und treten nur zum Datenaustausch zueinander in Kontakt. Zwischen den Datenaustauschphasen arbeiten der ASI-Master und der Hostrechner entkoppelt und autark.

In Fig. 3 ist die Aufteilung der Rechenzeit auf die beiden asynchronen Prozesse des ASI-Masters und des Hostrechners für den Fall gezeigt, daß der ASI-Steuermaster auf einem einzigen Controller realisiert ist.

In Fig. 3 bedeuten:

E/S = Ende eines Hostrechnerzyklusses und Start eines neuen Hostrechnerzyklusses

D = Datenaustausch zwischen ASI-Master und Hostrechner

A = Abarbeitung des Hostrechners

□ = Zeitscheiben des Hostrechners

Dem Hostrechner wird vom ASI-Steuermaster immer nur solange Rechenzeit zur Verfügung gestellt, wie der ASI-Master keine Rechenzeit beansprucht, also in den Lücken der Zykluszeit des ASI-Telegramms, wobei diese Lücken zeitlich unterschiedlich groß sein können. Die unterschiedlichen Längen dieser Zeitscheiben, die dem Hostrechner zur Verfügung gestellt werden, hängen vom ASI-Master ab.

In Fig. 4 ist der zyklische, zeitliche Ablauf des Hostprogramms des Hostrechners für den Fall gezeigt, daß der ASI-Steuermaster auf einem Controller realisiert ist. In der Fig. 4 ist der zyklische, zeitliche Ablauf des Hostprogramms, unter Berücksichtigung der unterschiedlich großen Zeitscheiben, die dem Hostrechner vom ASI-Master zur Verfügung gestellt werden, dargestellt. Die Zykluszeit des Hostrechners, also die Zeit zwischen dem Beginn von zwei Datenaustauschphasen, hängt zu einem großen Teil von der zur Verfügung gestellten Rechenzeit ab.

Fig. 5 zeigt ein mögliches Hauptdiagramm für den Datenaustausch beginnend mit dem Start des Datenaustauschs. Nach dem Start des Datenaustauschs wird abgefragt, ob ein ASI-Fehler aufgetreten ist; bei Bejahung wird der ASI-Kreis in den sicheren Zustand gefahren. Bei Verneinung wird der Datenaustausch durchgeführt. Danach wird wieder abgefragt, ob ein ASI-Fehler aufgetreten ist, bei Bejahung wird der ASI-Fehler in der Vergangenheit abgefragt und bei dessen Bejahung in einer Schleife zurück zur ersten Frage nach einem ersten ASI-Fehler gegangen. Bei Verneinung wird das Ende des Datenaustauschs angezeigt und vollzogen.

Ein sicherer und gefahrloser Zustand bei ASI ist beispielsweise die Offline-Phase. In der Offline-Phase wird zuerst an allen ASI-Slaves ein RESET durchgeführt und danach wird die Kommunikation auf der ASI-Leitung eingestellt. Nach dem RESET und dem Einstellen der Kommunikation besteht keine weitere Gefahr für eine Anlage.

Gewerbliche Anwendbarkeit

Der Gegenstand der Erfindung ist insbesondere für

Verfahren zur Steuerung und Aktivierung von miteinander mittels eines Bussystems vernetzten binären Sensoren und/oder Aktuatoren, ASI-Slaves, anwendbar, die von einem Verarbeitungsrechner, Hostrechner, wie speicherprogrammierbare Steuerung oder Bus-Rechner, über ein einen Controller aufweisendes Aktuator-Sensor-Interface, ASI-Master, angesteuert werden und umgekehrt, verwendbar, welches die auf das Bussystem von den ASI-Slaves aufgegebenen Signale des Hostrechners in einem vorgegebenen Zeitraster, ASI-Masterprogramm, zur Verfügung stellt, und der ASI-Master das Bussystem zu jedem Zeitpunkt in einen sicheren Zustand zu versetzen imstande ist. Die Nützlichkeit der Erfindung liegt insbesondere darin, daß der ASI-Steuermaster Steuerungsaufgaben übernimmt, für die bei konventioneller Realisierung des Standes der Technik ein Hostrechner zusätzlich erforderlich ist.

Patentansprüche

1. Verfahren zur Steuerung und Aktivierung von miteinander mittels eines Bussystems vernetzten binären Sensoren (6) und/oder Aktuatoren (ASI-Slaves), die von einem Verarbeitungsrechner (2) (Hostrechner), wie speicherprogrammierbare Steuerung oder Bus-Rechner, über ein einen Controller aufweisendes Aktuator-Sensor-Interface (1) (ASI-Master) angesteuert werden und umgekehrt, welches die auf das Bussystem (5, 6) von den ASI-Slaves (6) aufgegebenen Signale des Hostrechners (2) in einem vorgegebenen Zeitraster (ASI-Masterprogramm) zur Verfügung stellt, und der ASI-Master (1) das Bussystem (5, 6) zu jedem Zeitpunkt in einen sicheren Zustand zu versetzen imstande ist, dadurch gekennzeichnet, daß der Verarbeitungsrechner (2) als autarker Hostrechner innerhalb des ASI-Masters (1) integriert (ASI-Steuermaster (3)), aber getrennt aufgebaut ist und das ASI-Masterprogramm vom Programm des Hostrechners (Hostprogramm) entkoppelt ist, jedoch ein Datenaustausch zwischen beiden stattfindet, wobei das ASI-Masterprogramm des ASI-Masters immer Priorität gegenüber dem Hostprogramm des Hostrechners besitzt dergestalt, daß zu keinem Zeitpunkt der zeitliche Ablauf des ASI-Masters durch den Hostrechner bzw. das Hostprogramm verändert oder unterbrochen werden darf, hingegen der ASI-Master bzw. das ASI-Masterprogramm den Hostrechner bzw. das Hostprogramm zu jedem Zeitpunkt zu unterbrechen imstande ist und das Hostprogramm die Daten des Hostprogramms über eine interne Schnittstelle an den ASI-Master übergibt.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß innerhalb des ASI-Steuermasters die Prozeßdaten zweimal unabhängig voneinander abgelegt und verwaltet werden, wobei das ASI-Masterprogramm und das Hostprogramm immer nur auf seinen zugewiesenen eigenen Datenbereich zugreifen kann und die Daten der beiden Programme zu vorgebbaren Zeitpunkten ausgetauscht werden.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß bei einer Fehlererkennung durch den ASI-Master ein zusätzliches Fehlerbit gesetzt wird, welches nur vom ASI-Master gesetzt werden und nur vom Hostprogramm gelöscht bzw. quittiert werden darf, wobei durch das Speichern von Fehlern durch das Hostprogramm eine Konsistenz

der Daten beim Datenaustausch zwischen dem Hostrechner und dem ASI-Master erreicht wird.

4. Verfahren nach Anspruch 2, dadurch gekennzeichnet, daß vor dem Datenaustausch zwischen dem ASI-Masterprogramm und dem Hostprogramm überprüft wird, ob ein Fehler vorliegt und bei Nichtvorliegen eines Fehlers die Daten der beiden Programme ausgetauscht und als neue gültige Daten akzeptiert werden, hingegen bei Vorliegen eines Fehlers beide Prozesse in die jeweilige Fehlerbehandlung übergehen und die ausgetauschten Daten verwerfen.

5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß nach beendetem Datenaustausch erneut eine Fehlerprüfung durchgeführt wird, ob ein aktueller Fehler vorliegt oder ob in der Vergangenheit, angezeigt durch das ASI-Fehlerbit; zwischenzeitlich ein Fehler aufgetreten ist, wobei bis zum nächsten Datenaustausch beide Prozesse autark arbeiten.

6. Verfahren nach einem der vorherigen Ansprüche, dadurch gekennzeichnet, daß der ASI-Master bzw. das ASI-Masterprogramm die Zykluszeit des Hostprogramms ändern darf.

7. Verfahren nach einem der vorherigen Ansprüche, dadurch gekennzeichnet, daß lediglich der ASI-Master über eine eigene Schnittstelle nach außerhalb des ASI-Steuermasters Daten mit den ASI-Slaves austauscht, wobei der Hostrechner die Daten des Hostprogramms mit dem ASI-Master austauscht, der wiederum für den gesamten Datenaustausch mit den ASI-Slaves sorgt.

8. Verfahren nach einem der vorherigen Ansprüche, dadurch gekennzeichnet, daß der ASI-Kreis bei Fehlprogrammierung des Hostprogramms weiterhin spezifikationskonform betrieben wird, so daß diese Fehlprogrammierung den Ablauf des ASI-Masterprogramms des ASI-Masters nicht beeinflußt, wobei der ASI-Master den ASI-Kreis unabhängig vom Zustand des Hostprogramms in einen sicheren Zustand zu versetzen imstande ist.

9. Verfahren nach Anspruch 3, dadurch gekennzeichnet, daß der Hostrechner auf kurzzeitige ASI-Fehler in einer im Hostprogramm vorgegebenen Art und Weise zu reagieren imstande ist.

10. Vorrichtung zur Steuerung und Aktivierung von miteinander mittels eines Bussystems vernetzten binären Sensoren und/oder Aktuatoren (ASI-Slaves), die von einem Verarbeitungsrechner (Hostrechner), wie speicherprogrammierbare Steuerung oder Bus-Rechner, über ein einen Controller aufweisendes Aktuator-Sensor-Interface (ASI-Master) ansteuerbar sind und umgekehrt, welches die auf das Bussystem von den ASI-Slaves aufgegebenen Signale des Hostrechners in einem vorgegebenen Zeitraster (ASI-Masterprogramm) zur Verfügung stellt, und der ASI-Master das Bussystem zu jedem Zeitpunkt in einen sicheren Zustand zu versetzen imstande ist, dadurch gekennzeichnet, daß der Verarbeitungsrechner als autarker Hostrechner innerhalb des ASI-Masters (ASI-Steuermaster) integriert, aber getrennt aufgebaut ist und das ASI-Masterprogramm vom Hostprogramm des Hostrechners entkoppelt ist, jedoch ein Datenaustausch zwischen beiden stattfindet, wobei das ASI-Masterprogramm immer Priorität gegenüber dem Programm des Hostrechners besitzt dergestalt, daß zu keinem Zeitpunkt der zeitlich Ab-

lauf des ASI-Masters durch den Hostrechner bzw. das Hostprogramm verändert oder unterbrochen werden kann, hingegen der ASI-Master bzw. das ASI-Masterprogramm den Hostrechner bzw. das Hostprogramm zu jedem Zeitpunkt zu unterbrechen imstande ist und lediglich der ASI-Master eine Schnittstelle zum ASI-Kreis besitzt.

11. Vorrichtung nach Anspruch 10, dadurch gekennzeichnet, daß innerhalb des ASI-Steuermasters der interne ASI-Master und der interne Host auf ein- und demselben Controller realisiert sind.

12. Vorrichtung nach Anspruch 10, dadurch gekennzeichnet, daß innerhalb des ASI-Steuermasters der interne ASI-Master und der interne Host auf zwei verschiedenen Controllern realisiert sind.

Hierzu 3 Seite(n) Zeichnungen

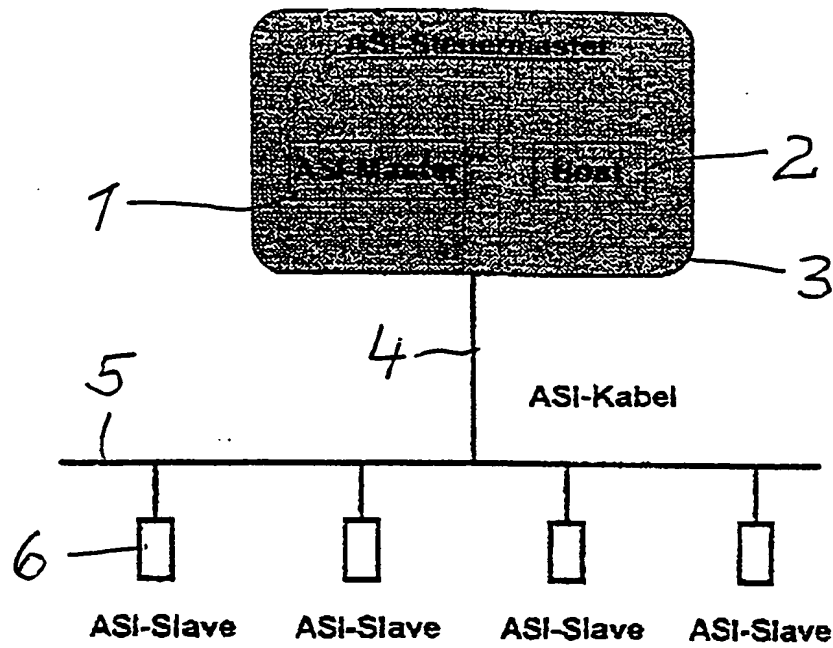


Fig. 1

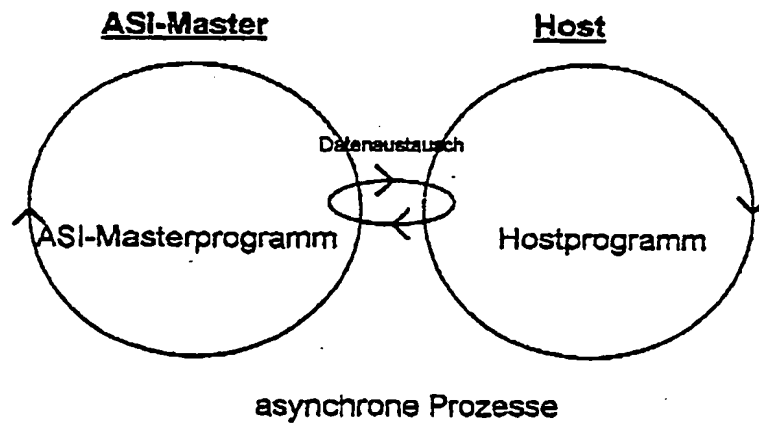


Fig. 2

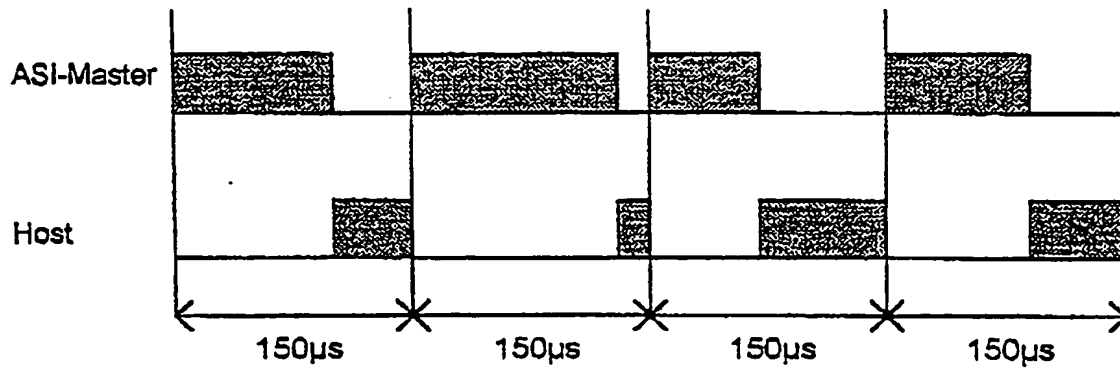


Fig. 3

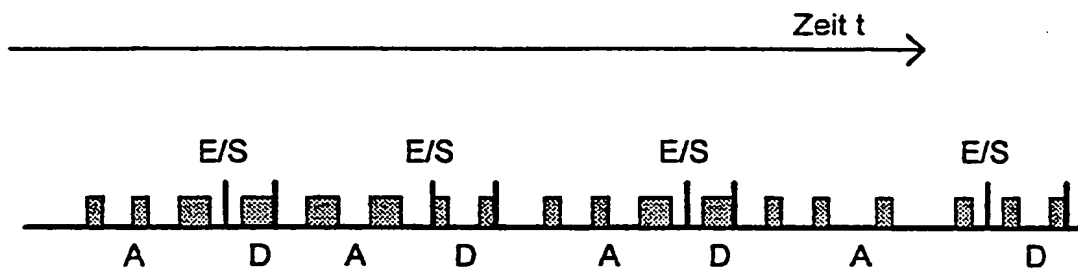


Fig. 4

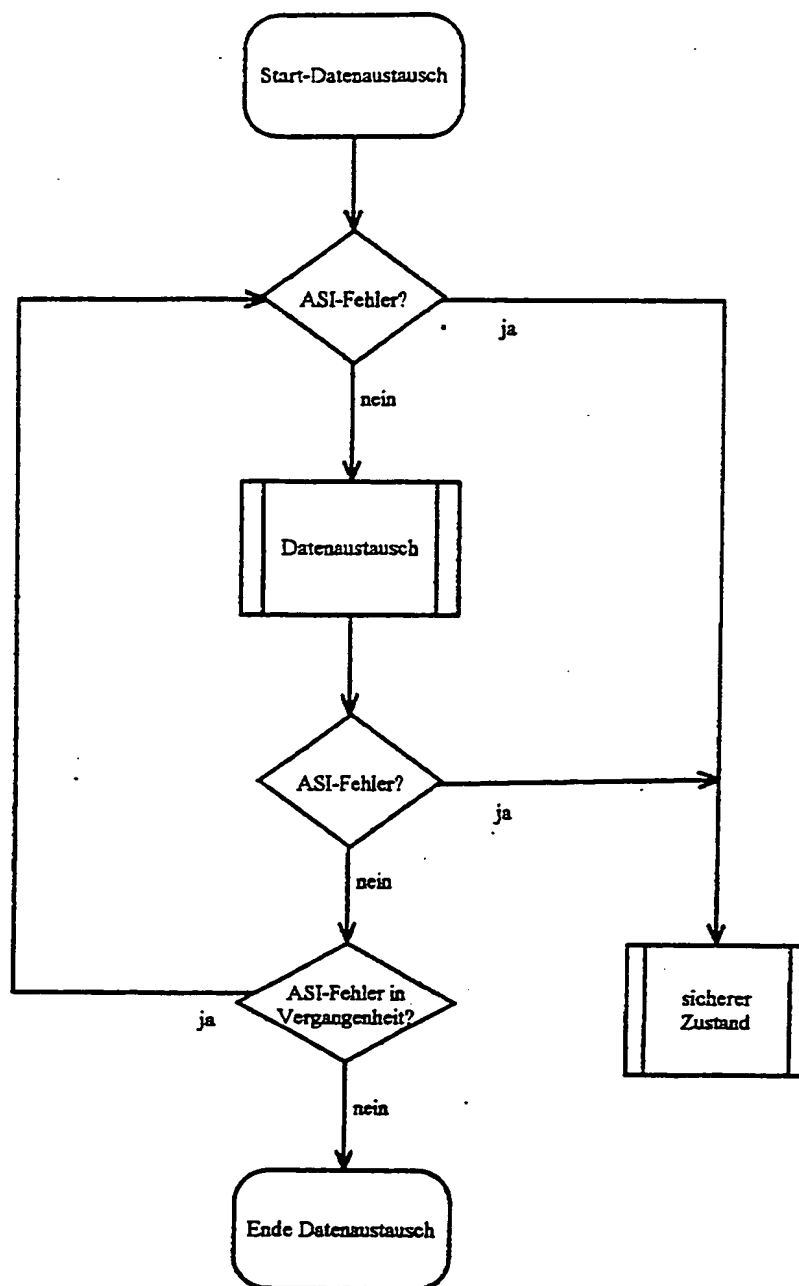


Fig. 5